

APPLICATION  
FOR  
UNITED STATES LETTERS PATENT

TITLE: IMPROVED DYNAMIC RANGE COMPRESSION OF AN  
AUDIO SIGNAL

APPLICANT: CLINTON A. STALEY

CERTIFICATE OF MAILING BY EXPRESS MAIL

Express Mail Label No EL228033698US

I hereby certify under 37 CFR §1.10 that this correspondence is being deposited with the United States Postal Service as Express Mail Post Office to Addressee with sufficient postage on the date indicated below and is addressed to the Commissioner for Patents, Washington, D.C. 20231.

Date of Deposit January 11, 2001

Signature Samantha Bell

Typed or Printed Name of Person Signing Certificate  
Samantha Bell

## 5

The invention relates to dynamic range compression of audio signals.

10

## 15

20

25

30

points on the trailing side of the range are dropped as the trailing edge of the range traverses the data points. Data points within the range are popped as they fall below the envelope being extended to data points at the leading edge of the range. The determination of the envelope of the amplitude of the audio signal has an order of

5 complexity of  $O(n)$ , where  $n$  is the number of audio signal values. The amplitude value for the audio signal is examined at most a fixed number of times in the determination of the envelope of the amplitude of the audio signal.

The control signal function generation includes using a first convex hull calculation for the MinMF values and a second convex hull calculation for the MaxMF

10 values to identify line segments of the control signal function that are between the MinMF and MaxMF values and have reduced slope. The first convex hull calculation results in a low hull, and the second convex hull calculation results in a high hull, and a segment of the control signal function is determined each time that the low hull and high hull cross. The MinMF and MaxMF values are converted to logarithms prior to using the

15 first and second convex hull calculations. Logarithms of the MinMF values and the MaxMF values are used to generate segments of a logarithm of the control signal function such that values of the logarithm of the control signal function are between the logarithm of MinMF and logarithm of MaxMF values, and such that segments of the logarithm of the control signal function have reduced slopes. The envelope has vertices,

20 and the creation of a control signal function has an order of complexity of  $O(n)$ , where  $n$  is the number of envelope vertices. Each vertex is examined at most a fixed number of times in the creating a control signal function.

In other aspects, the invention features, in general, using the control signal generated as just described to adjust audio signals, and apparatus creating and applying

25 the control signal to adjust audio signals.

In another aspect, the invention features, in general, creating a reduced-slope series of line segments passing through a pair of Max and Min limiting functions specifying  $y$  values with respect to a variable  $x$ . First, the Max and Min limiting functions are defined. Then a first convex hull calculation is used for  $y$  values of the Min

30 limiting function and a second convex hull calculation is used for  $y$  values of the Max

limiting function to identify line segments that are between the Min and Max values and have reduced slope.

Embodiments of the invention may include one or more of the following advantages. Audio samples are dynamically range compressed to values within a permitted dynamic range by gradual adjustments to the gain applied to the signal. Geometric adjustments are used to avoid adjustments that are perceptible to the listener. The adjustments have a linear complexity; that is, each audio data point is at most examined a fixed number of times regardless of the size of the audio file. Adjustments are as slow as possible consistent with guaranteeing that audio samples fall within the permitted dynamic range.

Other advantages and features of invention will be apparent from the following description of a preferred embodiment thereof and from the claims.

#### Description of the Drawings

Fig. 1 is a block diagram of a dynamic range compression device.

FIG. 2 is a graph of amplitude vs. time showing an audio signal and an envelope for the audio signal.

FIG. 3 is a graph of amplitude vs. time showing minimum and maximum multiplication factors for the FIG. 2 envelope and a control signal contained within the minimum and maximum multiplication factor tunnel.

FIGS. 4-8 illustrate calculation of the FIG. 3 control signal.

#### Detailed Description of Embodiment

Referring to FIG. 1, dynamic range compression unit 10 (which can be implemented in software or hardware) receives audio input 12 and applies a gain to it according to control signal (Cs) 14 in adjusting the audio output 16, which has minimum and maximum permitted amplitude values within a desired range. The control signal generated by Cs generator 18 provides gradual adjustments to the volume in advance of loud or quiet portions of the signal to bring the maximum and minimum volumes into range without noticeable adjustments.

Generator 18 employs a three-step procedure in order to provide gradual adjustments in generating the Cs values.

First a signal envelope 20 is created for the absolute values 22 of the input audio signal, as shown in FIG. 2. The envelope is used to prevent wave troughs in the signal from being interpreted as quiet portions requiring amplification to stay within the specified dynamic range.

Next there is calculated, for each point on the envelope, a minimum and maximum multiplication factor (MinMF and MaxMF) that can be multiplied times the envelope value such that the product is above a predetermined minimum amplitude and below a predetermined maximum amplitude of the dynamic range. These MinMF and MaxMF values are shown plotted on Fig. 3, and it is seen that they form a “VMR tunnel 24” (explained below) between them.

The last step is generation of a control signal Cs function 26 (FIG. 3) such that the values of Cs are within the tunnel 24, and the Cs function has small slopes, given the constraints of the tunnel. Keeping the Cs values within the MinMF/MaxMF tunnel guarantees that no audio sample is outside of the permitted dynamic range. Keeping the slopes small provides for gradual adjustments.

These steps of generating the Cs values are described in detail below.

#### Envelope Calculation

Referring to FIGS. 1 and 2, generator 18 (which is implemented in software) first computes the envelope 20 of the signal, by taking the absolute value of each audio point of the input signal and using a modified Graham's scan convex-hull algorithm (see Cormen, Introduction to Algorithms, ISBN 0-07-013143-0, pp. 898-905) to generate a “convex hull” across the top of the absolute values 22. Graham's algorithm is typically used to create a convex hull for a set of two-dimensional coordinates. The algorithm considers each point in the set in sequence as a potential member of the set of points on the convex hull, and “pops” (i.e., removes) a point from the convex hull when it appears that including it will result in a concave portion on the convex hull. Each point is considered in rotational order with respect to a starting point.

In computing envelope 20 for the audio samples, the typical Graham algorithm is modified by visiting points in left-to-right order instead of rotational order, and by using a moving set containing a fixed number of audio sample points that traverses the entire audio program instead of considering the entirety of the set of points at one time. The moving set is held in a so-called “deque” (a queue that can be added to or deleted from on either end). The size  $R$  of the moving set (in number of samples) should be greater than or equal to half the longest wavelength to avoid interpreting the rise and fall of sinusoids as volume peaks to be damped. E.g.,  $R$  could be 750 samples (or greater), given a longest wavelength of 1/30 second and a sampling rate of 44KHz.

Referring to Fig 2, the current envelope generation procedure maintains a set of “potential envelope points,” composed either of actual audio samples, or of “generated envelope points” as described below. At any point in the process (after initialization), a deque  $Q$  has a set of up to  $R$  potential envelope points spanning exactly  $R$  samples. As the hull computation advances rightward to each successive sample  $S$ , generator 18 drops off the left end of  $Q$  any samples that are not from the  $R$  prior samples. Samples that are so dropped are considered a permanent part of the envelope.

The rightmost potential envelope point, at sample  $T$ , is always an actual audio sample. The leftmost potential envelope point, at  $T-R+1$ , may be an actual audio sample. However, the actual audio sample at  $T-R+1$  may have been removed from  $Q$  earlier to eliminate a concavity. In this event, generator 18 maintains at  $T-R+1$  a generated envelope point derived by interpolating linearly between the closest audio samples that still remain on the envelope. There are always potential envelope points at  $T$  and  $T-R+1$ , though depending on the removal of concavities these may be the only points in  $Q$ .

To advance the envelope calculation, generator 18 successively adds new audio samples to the right end of  $Q$ , uses the Graham-scan comparison of the last two slopes on the envelope to remove any potential envelope points within  $Q$  that cause concavities with the newly added point, and drops one potential envelope point from the left end of  $Q$  to maintain the range of  $R$ . All potential envelope points dropped from the left end of  $Q$  are no longer subject to removal due to concavities, and thus become permanent envelope points, with one exception. The algorithm may generate a series of permanent envelope

points that are collinear, and to save space and time all but the first and last permanent envelope points in such series are removed.

The following pseudocode describes the envelope generation process.

```

5  set deque Q to empty
  for each sample absolute value  $S_t$  {
    If  $S_t$  is below minimum threshold, set  $S_n$  to minimum threshold.
    Add  $S_t$  to Q
    while more than two samples exists in Q, and
10   the angle between the final two line-segments in Q is concave {
      drop the second-to-rightmost point from Q
    }
    if the range of envelope points in Q exceeds R {
      Drop the leftmost point LP from Q, adding it to the permanent envelope
15   Remove from the permanent envelope any points to the left of LP that are
      collinear with LP and with the point to their left.
      Create a new generated envelope point at  $T-R+1$  if necessary
    }
  }
20   In generating envelope 20, a minimum threshold amplitude is set. Any input
  audio values below this threshold are assigned the minimum threshold value. This avoids
  attempting to bring a signal amplitude of 0 to some finite value via multiplication times
  Cs. E.g., a minimum threshold could be set to 1% of maximum volume. In practice,
  setting minimum threshold to the same value as the minimum dynamic volume works
25  best.

  Owing to the large size of audio files, they are typically broken down into
  segments, e.g., of 8192 bytes, and volume adjustments are computed and returned for one
  segment before beginning the calculations for the next segment. The envelope
  computation and the subsequent computations below are all implemented for one
30  segment before starting the next segment. This necessitates a look ahead of R samples
  into the next segment, however, so that Q may be advanced to the point where the final

```

time point of the current segment has been dropped from Q's left end and added to the permanent envelope.

### Multiplication Factor Tunnel Generation

After the envelope has been computed, the envelope itself is used as the signal to be range-limited. Note in particular that if the envelope is kept below a certain maximum value, then the absolute value of all of the original audio samples will also be kept below that value, by the convex-hull property of the envelope.

Referring to FIG. 3, for every point on the envelope, there is a volume multiplication range (VMR) formed by the minimum and maximum multiplication factors (MinMF, MaxMF) that can be multiplied times the amplitude value to keep the amplitude within the permitted min/max volume range. MinMF and MaxMF are calculated by the following formulas:

MinMF = minimum permitted amplitude/envelope value

MaxMF = maximum permitted amplitude/envelope value

The MinMF and MaxMF functions 28, 30 form VMR tunnel 24 as shown on FIG. 3. For a signal that needs no adjustment, the horizontal line  $y = 1$  (or  $y = 0$  if logarithmic space is used as discussed below) will always lie within VMR tunnel 24. For signals that need amplification, the bottom of VMR tunnel 24 (the MinMF function 28) will lie above the  $y=1$  line, and for signals that need deamplification, the top of the VMR tunnel (the MaxMF function 28) will lie below the  $y=1$  line.

As noted above, segments of the audio file are considered in sequence; thus, in the processing of a segment of the audio file, after the MinMF and MaxMF values have been calculated for that segment, subsequent processing is performed on that segment before calculating the envelope and MinMF and MaxMF values for the next segment.

### Control Signal Generation

Still referring to FIG. 3, the control signal function 26 should have the smallest possible slope to avoid conspicuous volume adjustments. While the function could include segments of straight lines kept within tunnel 24, that would result in linear volume adjustments; however, because geometric volume adjustments sound more



uniform to the ear, they are preferred. To do this, the original audio points are converted to logarithmic form before envelope computation. The resultant MinMF and MaxMF data points are thus also in logarithmic form, and the logarithm of the control signal CS function 26 is generated by identifying a set of lines with the smallest slopes that fit

5 within the logarithmic MinMF and MaxMF functions. This set of lines in logarithmic space translates to geometric or exponential curves in normal space. Note also that if the logarithms of individual CS points do not protrude above the envelope in logarithmic space, they will not protrude above the translated envelope (a series of exponential curves instead of a series of line segments) in normal space, since log is a monotonic function.

10 Referring to FIG. 4, logarithmic-space VMR tunnel 32 has a fixed width and polygonal sides for the logarithmic MinMF function 34 and logarithmic MaxMF function 36. The vertices of log MinMF 34 and log MaxMF 36 are derived from the vertices of the signal envelope, and thus come in pairs with the same x-value. These are termed “VMR vertex pairs.”

15 Still referring to FIG. 4, two convex hulls 38, 40, are employed, using another modification of Graham's scan (referenced earlier), to determine the logarithm of the control signal function 26. In this case, vertex pairs are considered (instead of values for each audio sample point). Also, instead of using a moving range R in which values to the left are dropped after the range passes over them, in this case the values are removed

20 from the scan at the left each time that hulls 38, 40 cross, as described below. Hull 38 is the low hull, which is stretched across log MinMF 34; hull 40 is the high hull, which is stretched across log MaxMF 36. Both hulls 38, 40 start at a common hull base point 42, initially set to the middle of the logarithmic VMR tunnel 32, at the left hand side. The high and low points of successive VMR vertex pairs are then considered in advancing the

25 hulls 38, 40, respectively, to the right. As each new point LP from curve 34 is added to hull 38, any points at the end of the hull that would form an upward-facing concavity with LP added are first removed. This is done by the Graham-scan approach: repeatedly examining the last two segments in hull 38 for slopes indicating an upward concavity, and if necessary removing the next-to-last point from the hull 38 to remove the concavity,

30 until there is no upward concavity. Note that in some cases this may result in the removal

of all points in hull 38 but the leftmost. When this happens, the slope of the initial segment in hull 38 increases by some amount. This is called a “narrowing” of the hull. Similarly, as each new point HP from curve 36 is added to hull 40, any points at the end of the hull that would form a downward-facing concavity with HP added are first  
 5 removed. If all but the leftmost point are removed, then hull 40 is “narrowed”, with the slope of its initial line segment decreasing.

As a result of this modified Graham-scan process, the first (leftmost) segment of the high hull 40 in FIG. 4 may (and almost always will) take on a lower and lower slope due to repeated narrowings. Similarly, the first segment of hull 34 takes on a higher  
 10 slope with each narrowing.

After some number of narrowings, the first segment of the low hull will have higher slope than the first segment of the high hull. This is considered a “hull crossing.” E.g., FIG.5 shows the high hull's first segment having a lower slope than the low hull's first segment. When the hulls cross, one, and only one, of the hulls has a single segment  
 15 directly from the hull base to the high or low vertex of the just-added VMR vertex pair; this hull is considered the simple hull. The other hull must have more than one segment, and is considered the compound hull. When the hulls cross, the leftmost segment of the compound hull is designated as a permanent segment of the log control signal function 46 (FIG.6; the log of control signal function 26), and is removed from consideration in the  
 20 ongoing advancement to the right. The end of this segment becomes the new hull base point 48 of both hulls, and changes the starting point of the first segment of the simple hull and the compound hull, as is shown in FIG. 6 If the hulls still cross after this adjustment, we repeat the same adjustment, removing further leftmost segments from the compound hull and adding the removed segments to the log control signal function 46,  
 25 until the hulls no longer cross. This must happen before all segments are exhausted from the compound hull, since if each hull has only one segment remaining, those segments must lead from the hulls' common base point to the high and low points of the most-recently added VMR vertex pair, and the hulls do not cross. Addition of VMR vertex pairs to the hulls continues, with new additions to the log control signal function 46  
 30 occurring each time the hulls cross. At each point in the process, there are completed segments of the log control signal function 46 up to (to the left of) the new hull base 48,

non-crossing low and high hulls 38, 40 to the right of the new hull base 48 and to the left of the most-recently added VMR vertex pair, and unexplored areas of the logarithmic VMR tunnel 32 to the right of the most-recently added VMR vertex pair.

As noted above, audio files are typically broken down into segments, using a limited look ahead at the data beyond the end of the current segment. In processing by segments, the values of log control signal function 46 must be computed for the current segment, even if the look ahead data is not sufficient to cause hull crossings that would generate the values for log control signal 46 in the normal way. When the full look ahead for a given segment has been used without obtaining values for log control signal function 46 that cover the entire segment, the processing is in an "indeterminate slope state" such as illustrated in FIG. 7. As shown in FIG. 7, high hull 40 and low hull 38 do not cross in either current segment 50 or look ahead segment 52. To resolve an indeterminate slope state, a slope between the first segments of the high and low hulls is selected for log control signal 46 for the current segment 50, as shown in FIG.8. The slope may be selected to force the log multiplier back to 0 as much as possible, or it may be selected to minimize the slope of log control signal function 46. The ending point 54 of the line of the selected slope becomes the new hull base for the next segment 56.

This process produces a log control signal function 46 with the smallest overall slope while still staying within the log VMR tunnel 32 (with the possible exception of slopes chosen to resolve indeterminate slope states between audio segments). The order of complexity of this process is  $O(n)$ , with  $n$  being the number of envelope vertices. In the hull computations, each point is processed at most twice -- once to be added to the hull, and to be removed. The per-point processing thus takes an amount of time independent of the number of points overall. The  $O(n)$  complexity of the described algorithm results in very fast execution.

Once the full logarithmic control signal function 46 has been computed for all audio segments, the antilog of the y-value of the log control signal function is taken at each audio sample point (each x point) to obtain the values in control signal function 26 of the control signal  $C_s$  for each audio sample. Respective values of  $C_s$  are then multiplied times the audio input signal values, resulting in an output signal with minimal, geometric, volume adjustments that maintain an amplitude within the specified range.

In multiplying the Cs values times the input audio values, input audio values below the threshold mentioned earlier are assigned to the minimum threshold value to avoid attempting to bring a signal amplitude of 0 to some finite value via multiplication times Cs. The original audio data, however, is not changed. Best results are obtained by  
5 using the same value for the minimum amplitude and the threshold. This prevents spurious amplification of very-quiet signals that lie between the minimum amplitude and the threshold.

A stored audio program can be accessed and processed by generator 18 and unit  
10 and then stored in dynamically range-compressed form. Audio programs being input or output from audio equipment can also be dynamically range compressed on the fly, e.g., in video or audio streaming applications or in the creation of compressed video or audio programs.

Other embodiments of the invention are within the scope of the following claims:

15 What is claimed is: